```c
/******************************************************************************

                              Online C Compiler.
                Code, Compile, Run and Debug C program online.
Write your code in this editor and press "Run" button to compile and execute

*******************************************************************************


#include<stdio.h>
#include<ctype.h>

#define SIZE 50

char stack[SIZE];
int top=-1;

void push(char elem)
{
    stack[++top]=elem;
}

char pop()
{
    return stack[top--];
}

int pr(char symbol)
{
    if(symbol=='^')
    {
        return(3);
    }
    else if(symbol=='*'||symbol=='/')
    {
        return(2);
    }
    else if(symbol=='+'||symbol=='-')
```

```c
38      else if(symbol=='+'||symbol=='-')
39      {
40          return(1);
41      }
42      else
43      {
44          return(0);
45      }
46  }
47
48  int main()
49  {
50      char infix[50],postfix[50],ch,elem;
51      int i=0,k=0;
52
53      printf("Enter the Infix expression: ");
54      scanf("%s",&infix);
55
56      push('#');
57
58      while((ch=infix[i++])!='\0')
59      {
60          if(ch=='(')push(ch);
61          else
62              if(isalnum(ch))postfix[k++]=ch;
63              else
64                  if(ch==')')
65                  {
66                      while(stack[top]!='(')
67                          postfix[k++]=pop();
68                      elem=pop();
69                  }
70                  else
71                  {
72                      while(pr(stack[top])>=pr(ch))
73                          postfix[k++]=pop();
74                      push(ch);
75                  }
```

```c
47
48   int main()
49   {
50       char infix[50],postfix[50],ch,elem;
51       int i=0,k=0;
52
53       printf("Enter the Infix expression: ");
54       scanf("%s",&infix);
55
56       push('#');
57
58       while((ch=infix[i++])!='\0')
59       {
60           if(ch=='(')push(ch);
61           else
62               if(isalnum(ch))postfix[k++]=ch;
63               else
64                   if(ch==')')
65                   {
66                       while(stack[top]!='(')
67                           postfix[k++]=pop();
68                       elem=pop();
69                   }
70                   else
71                   {
72                       while(pr(stack[top])>=pr(ch))
73                           postfix[k++]=pop();
74                       push(ch);
75                   }
76       }
77       while(stack[top]!='#')
78           postfix[k++]=pop();
79
80       postfix[k]='\0';
81       printf("\nPostfix Expression = %s\n",postfix);
82
83       return 0;
84   }
```

```
Enter the Infix expression: (a+b)*e/(c-d)

Postfix Expression = ab+e*cd-/


...Program finished with exit code 0
Press ENTER to exit console.
```